

## CLAIMS

What is claimed is:

- 1 1. A method of processing a request for a count-limited network resource, the method  
2 comprising the computer-implemented steps of:  
3 receiving a request from a client process for a computer resource in a network;  
4 determining whether the request exceeds a maximum count of allowed requests for  
5 the resource, and if so,  
6 determining whether a current time is within a retry time period of a first time  
7 of a first request for the resource by the client process, and if so,  
8 automatically returning to the step of determining whether the request  
9 exceeds the maximum count for the resource; and  
10 if the request does not exceed the maximum count, then providing the resource to the  
11 client process.
- 1 2. A method as recited in Claim 1, further comprising the steps of determining whether a  
2 request from the client process to set the retry time period has been received, and if  
3 so, automatically repeating the repeating and determining steps of Claim 1.
- 1 3. A method as recited in Claim 2, further comprising receiving from the client process a  
2 request to set the retry time period and a data value indicating the retry time period.
- 1 4. A method as recited in Claim 3, further comprising setting the retry period based on  
2 the data value indicating the retry period.
- 1 5. A method as recited in Claim 1, wherein the retry period is zero seconds.

6. A method as recited in Claim 1, wherein said step of automatically returning to the step of determining whether the request exceeds the maximum count for the resource further comprises the step of waiting a predetermined delay period of time before returning.

7. A method as recited in Claim 1, wherein the step of determining whether the request exceeds the maximum count for the resource further comprises the steps of responding to a request for the resource associated with a first user having a first user identification associated with a first host that participates in a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts, and wherein a user identification for each user is associated with one host of the plurality of hosts by requesting a lock from a first local lock manager process executing on a first host.

8. A method of controlling use by concurrent users of a distributed resource on a network, wherein use of the resource is limited to a specified maximum number of concurrent users, the method comprising the computer-implemented steps of:

- providing a distributed lock manager process comprising a plurality of local lock manager processes executing on a corresponding plurality of hosts;
- associating a user identification for each user with one host of the plurality of hosts;
- and
- responding to a request for the resource associated with a first user having a first user identification associated with a first host of the plurality of hosts by requesting a lock from a first local lock manager process executing on the first host.

9. A method as recited in Claim 8, wherein during said step of associating a user identification with one host, the first user is associated with the first host based on information indicating that the first user signs onto the network most frequently at a terminal node of the network that uses fewer intervening network devices to pass messages to the first host than to any other host of the plurality of hosts.

10. A method as recited in Claim 8, wherein during said step of associating a user identification with one host, the first user is associated with the first host based on information indicating that the first user signs onto the network most frequently at a terminal node of the network geographically closer to the first host than to any other of the plurality of hosts.

11. A method as recited in Claim 8, wherein each local lock manager process maintains a lock data structure associated with the resource and the lock data structure includes a local resource maximum field; and further comprising the steps of generating and storing a value in the local resource maximum field maintained by each local lock manager process such that a summation over all the local lock manager process of the value in the local resource maximum field does not exceed the maximum number of concurrent users.

12. A method as recited in Claim 11, wherein a first value in the local resource maximum field maintained by the first local lock manager process is based on a number of users associated with the first host.

13. A method as recited in Claim 8, wherein a copy of the distributed resource resides on a host of the plurality of hosts.

14. A method as recited in Claim 8, wherein a copy of the distributed resource resides on a computing device that is not among the plurality of hosts.

15. A method as recited in Claim 11, further comprising:  
determining whether a number of outstanding locks granted by the first local lock manager process is less than a particular value stored in the local resource maximum field maintained by the first local lock manager process; and  
if the number of outstanding locks is less than the particular value, then generating and returning a lock object.

1 16. A method of controlling concurrent users of a distributed resource on a network, the  
2 resource limited to a maximum number of concurrent users, the method comprising the  
3 computer-implemented steps of:  
4 receiving a request for the distributed resource from a client process for a user having  
5 a user identification;  
6 determining a home location associated with the user identification, the home location  
7 indicating a unique host among a plurality of hosts that execute a  
8 corresponding plurality of local lock manager processes of a distributed lock  
9 manager process;  
10 sending a request for a lock object for the distributed resource to a first local lock  
11 manager process of the distributed lock manager process, the request including  
12 the home location;  
13 receiving the lock object for the distributed resource from a second local lock  
14 manager process executing on the unique host, if a number of outstanding  
15 locks granted by the second local lock manager process is less than a value of  
16 a local resource maximum defined for the second local lock manager process;  
17 and  
18 providing access to the resource to the first client only in response to receiving the  
19 lock object.

1 17. A method as recited in Claim 16, wherein a first host on which the first local lock  
2 manager process executes is different from the unique host on which the second local lock  
3 manager process executes.

1 18. A method as recited in Claim 16, wherein a first host on which the first local lock  
2 manager process is executing is in a local network with a computing device on which the  
3 client process is executing.

1 19. A method as recited in Claim 16, further comprising the steps of determining whether  
2 a request from the client process to set the retry time period has been received, and if so,  
3 automatically repeating the steps of determining, sending, receiving and providing.

1 20. A method of controlling concurrent users of a distributed resource on a network, the  
2 distributed resource limited to a maximum number of concurrent users, the method  
3 comprising the steps of:  
4 receiving at a first local lock manager process of a distributed lock manager process a  
5 request for a lock object for the distributed resource from a resource server,  
6 wherein the request includes data indicating a particular user home location;  
7 determining whether a second local lock manager process of the distributed lock  
8 manager process is associated with the particular user home location, and if so,  
9 requesting the lock object from the second local lock manager process.

1 21. A method as recited in Claim 20, further comprising the steps of:  
2 when the second lock manager process is not associated with the particular home  
3 location, determining whether a first number of outstanding locks granted by  
4 the first local lock manager process for the distributed resource is less than a  
5 first local maximum number of users no greater than the maximum number of  
6 concurrent users, and if so, then incrementing the first number of outstanding  
7 locks granted and providing the lock object to the resource server.

1 22. A method as recited in Claim 20, wherein each local lock manager process of the  
2 distributed lock manager process stores a corresponding local maximum number of users for  
3 the distributed resource in a lock data structure associated with the local lock manager  
4 process, and an aggregate of local maximum numbers in lock data structures associated with  
5 all lock manager processes in the distributed lock manager process for the distributed  
6 resource is no greater than the maximum number of concurrent users to which the distributed  
7 resource is limited.

1 23. A method as recited in Claim 20, further comprising the steps of:  
2 receiving a request for a lock object from a third local lock manager process of the  
3 distributed lock manager process in response to a request from a second  
4 resource server;  
5 determining whether a first number of outstanding locks granted by the first lock  
6 manager process for the distributed resource is less than a first local maximum  
7 number of users no greater than the maximum number of concurrent users;  
8 and  
9 if it is determined the first number of outstanding locks granted is less than the first  
10 local maximum number, then  
11 incrementing the first number of outstanding locks granted, and  
12 providing the lock object for the second resource server.

1 24. A method as recited in Claim 20, further comprising the step of providing the lock  
2 object to the resource server in response to receiving the lock object from the second lock  
3 manager process.

1 25. A method of distributing a resource on a network, the resource limited to a maximum  
2 number of concurrent users, the method comprising the steps of:  
3 providing a distributed lock manager process comprising a plurality of local lock  
4 manager processes executing on a corresponding plurality of hosts;  
5 generating a value for a local resource maximum number of users stored on each host  
6 of the plurality of hosts such that a summation over the plurality of hosts of  
7 the value for the local resource maximum yields an aggregate value that does  
8 not exceed the maximum number of concurrent users;  
9 determining whether to increase a first value in a first resource maximum stored on a  
10 first host of the plurality of hosts; and  
11 if it is determined to increase the first resource maximum, then  
12 decreasing by a particular amount a second value in a second resource  
13 maximum stored on a second host of the plurality of hosts, and

14 increasing by the particular amount the first value in the first resource  
15 maximum stored on the first host,  
16 wherein each local lock manager process is configured to grant a lock for the resource  
17 if the number of outstanding locks granted by the local lock manager process  
18 is less than a value of the local resource maximum stored on the corresponding  
19 host.

1 26. A method as recited in Claim 25, said step of determining whether to increase the first  
2 value further comprising determining the time of day and the geographic location served by  
3 the first host.

1 27. A method as recited in Claim 25, said step of determining whether to increase the first  
2 value further comprising determining whether a difference between the number of  
3 outstanding locks granted and the first value is less than a first predetermined threshold.

1 28. A method as recited in Claim 25, further comprising, if it is determined to increase the  
2 first value, then determining whether to decrease the value based on a whether a difference  
3 between the number of outstanding locks granted and the second value is greater than a  
4 second predetermined threshold.

1 29. A method as recited in Claim 28, wherein the particular amount is based on the  
2 second predetermined threshold.

1 30. A method as recited in Claim 29, wherein the particular amount is no greater than the  
2 second predetermined threshold.

31. A computer-readable medium carrying one or more sequences of instructions for handling a request for a count-limited resource, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

- receiving a request from a client process for a computer resource in a network;
- determining whether the request exceeds a maximum count of allowed requests for the resource, and if so,
- determining whether a current time is within a retry time period of a first time of a first request for the resource by the client process, and if so, automatically returning to the step of determining whether the request exceeds the maximum count for the resource; and
- if the request does not exceed the maximum count, then providing the resource to the client process.

32. A computer-readable medium carrying one or more sequences of instructions for controlling a number of concurrent users of a distributed resource on a network, the resource limited to a maximum number of concurrent users, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

- associating a user identification for each user with one host of a plurality of hosts on which are executing a corresponding plurality of local lock manager processes of a distributed lock manager process; and
- responding to a request for the resource associated with a first user having a first user identification associated with a first host of the plurality of hosts by requesting a lock from a first local lock manager executing on the first host.

33. A computer-readable medium carrying one or more sequences of instructions for controlling a number of concurrent users of a distributed resource on a network, the resource limited to a maximum number of concurrent users, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

- receiving a request for the distributed resource from a client process for a user having a user identification;



7 determining one home location associated with the user identification, the home  
8 location indicating a unique host among a plurality of hosts executing a  
9 corresponding plurality of local lock manager processes of a distributed lock  
10 manager process; and  
11 sending a request for a lock object for the distributed resource to a first local lock  
12 manager process of the distributed lock manager process, the request including  
13 the home location;  
14 receiving the lock object for the distributed resource from a second local lock  
15 manager process executing on the unique host, if a number of outstanding  
16 locks granted by the second local lock manager process is less than a value of  
17 a local resource maximum defined for the second local lock manager process;  
18 and  
19 providing access to the resource to the first client in response to receiving the lock  
20 object.

1 34. A computer-readable medium carrying one or more sequences of instructions for  
2 controlling a number of concurrent users of a distributed resource on a network, the  
3 distributed resource limited to a maximum number of concurrent users which instructions,  
4 when executed by one or more processors, cause the one or more processors to carry out the  
5 steps of:  
6 receiving at a first local lock manager process of a distributed lock manager process a  
7 request for a lock object for the distributed resource from a resource server,  
8 the request including data indicating a particular user home location, and the  
9 first local lock manager process executing on a first host;  
10 determining whether a second local lock manager process of the distributed lock  
11 manager process is associated with the particular user home location, the  
12 second local lock manager process executing on a second host different from  
13 the first host; and  
14 if it is determined that the second local lock manager process is associated with the  
15 particular home location, then requesting the lock object from the second local  
16 lock manager process.

1 35. A computer-readable medium carrying one or more sequences of instructions for  
2 distributing a resource on a network, the resource limited to a maximum number of  
3 concurrent users, which instructions, when executed by one or more processors, cause the one  
4 or more processors to carry out the steps of:  
5 generating a value for a local resource maximum number of users stored on each host  
6 of a plurality of hosts executing a corresponding plurality of local lock  
7 manager processes of a distributed lock manager process such that a  
8 summation over the plurality of hosts of the value for the local resource  
9 maximum yields an aggregate value that does not exceed the maximum  
10 number of concurrent users;  
11 determining whether to increase a first value in a first resource maximum stored on a  
12 first host of the plurality of hosts; and  
13 if it is determined to increase the first resource maximum, then  
14 decreasing by a particular amount a second value in a second resource  
15 maximum stored on a second host of the plurality of hosts, and  
16 increasing by the particular amount the first value in the first resource  
17 maximum stored on the first host,  
18 wherein each local lock manager process is configured to grant a lock for the resource  
19 if the number of outstanding locks granted by the local lock manager process  
20 is less than a value of the local resource maximum stored on the corresponding  
21 host.

1 36. An apparatus for handling a request for a count-limited resource, comprising:  
2 means for receiving a request from a client process for a computer resource in a  
3 network;  
4 means for determining whether the request exceeds a maximum count of allowed  
5 requests for the resource, and if so, for determining whether a current time  
6 is within a retry time period of a first time of a first request for the resource  
7 by the client process, and if so, for automatically returning to the step of  
8 determining whether the request exceeds the maximum count for the  
9 resource; and if the request does not exceed the maximum count, for  
10 providing the resource to the client process.

1 37. An apparatus for controlling a number of concurrent users of a distributed resource on  
2 a network, the resource limited to a maximum number of concurrent users, comprising:  
3 means for associating a user identification for each of the concurrent users with a  
4 unique host; and  
5 means for responding to a request for the resource associated with a first user among  
6 the concurrent users who has a first user identification associated with a first  
7 host by requesting a lock from a local lock manager process executing on the  
8 first host.

1 38. An apparatus for controlling a number of concurrent users of a distributed resource on  
2 a network, the resource limited to a maximum number of concurrent users, comprising:  
3 means for receiving a request for the distributed resource from a client process for a  
4 user having a user identification;  
5 means for determining one home location associated with the user identification, the  
6 home location indicating a unique host among a plurality of hosts executing a  
7 corresponding plurality of local lock manager processes of a distributed lock  
8 manager process;  
9 means for sending a request for a lock object for the distributed resource to a first  
10 local lock manager process of the distributed lock manager process, the  
11 request including the home location;

means for receiving the lock object for the distributed resource from a second local lock manager process executing on the unique host, if a number of outstanding locks granted by the second local lock manager process is less than a value of a local resource maximum defined for the second local lock manager process; and  
means for providing access to the resource to the first client in response to receiving the lock object.

39. An apparatus for controlling a number of concurrent users of a distributed resource on a network, the distributed resource limited to a maximum number of concurrent users, comprising:

means for receiving at a first local lock manager process of a distributed lock manager process a request for a lock object for the distributed resource from a resource server, the request including data indicating a particular user home location, and the first local lock manager process executing on a first host;

means for determining whether a second local lock manager of the distributed lock manager is associated with the particular user home location, the second local lock manager process executing on a second host different from the first host; and

means for requesting the lock object from the second local lock manager, if it is determined that the second local lock manager is associated with the particular home location.

40. An apparatus for distributing a resource on a network, the resource limited to a maximum number of concurrent users, comprising:

means for generating a value for a local resource maximum number of users stored on each host of a plurality of hosts in the network such that a summation over the plurality of hosts of the value for the local resource maximum yields an aggregate value that does not exceed the maximum number of concurrent users;

8 means for determining whether to increase a first value in a first resource maximum  
9 stored on a first host of the plurality of hosts; and  
10 means for decreasing by a particular amount a second value in a second resource  
11 maximum stored on a second host of the plurality of hosts, if it is determined  
12 to increase the first resource maximum;  
13 means for increasing by the particular amount the first value in the first resource  
14 maximum stored on the first host, if it is determined to increase the first  
15 resource maximum;  
16 means for granting a lock for the resource if the number of outstanding locks granted  
17 by the local lock manager is less than a value of the local resource maximum  
18 stored on the corresponding host.

41. An apparatus for processing a request for a count-limited network resource,  
comprising:  
a host that is communicatively coupled to a network that contains the count-limited  
network resource and that comprises one or more processors and a computer-  
readable storage medium;  
one or more sequences of instructions stored in the computer-readable storage  
medium which, when executed by the one or more processors, cause the one  
or more processors to carry out the steps of:  
receiving a request from a client process for a computer resource in a network;  
determining whether the request exceeds a maximum count of allowed  
requests for the resource, and if so,  
determining whether a current time is within a retry time period of a first time  
of a first request for the resource by the client process, and if so,  
automatically returning to the step of determining whether the request  
exceeds the maximum count for the resource; and  
if the request does not exceed the maximum count, then providing the resource  
to the client process.

1 42. An apparatus for controlling use by concurrent users of a distributed resource on a  
2 network, wherein use of the resource is limited to a specified maximum number of concurrent  
3 users, comprising:

4 a host that is communicatively coupled to a network that contains the distributed  
5 resource and that comprises one or more processors and a computer-readable  
6 storage medium;

7 one or more sequences of instructions stored in the computer-readable storage  
8 medium which, when executed by the one or more processors, cause the one  
9 or more processors to carry out the steps of:

10 associating a user identification for each user with one host of the plurality of  
11 hosts; and

12 responding to a request for the resource associated with a first user having a  
13 first user identification associated with a first host of the plurality of  
14 hosts by requesting a lock from a lock manager process executing on  
15 the first host.

1 43. An apparatus controlling concurrent users of a distributed resource on a network,  
2 wherein the resource is limited to a maximum number of concurrent users, comprising:

3 a host that is communicatively coupled to a network that contains the distributed  
4 resource and that comprises one or more processors and a computer-readable  
5 storage medium;

6 one or more sequences of instructions stored in the computer-readable storage  
7 medium which, when executed by the one or more processors, cause the one  
8 or more processors to carry out the steps of:

receiving a request for the distributed resource from a client process for a user having a user identification;  
determining a home location associated with the user identification, the home location indicating a unique host among a plurality of hosts that execute a corresponding plurality of local lock manager processes of a distributed lock manager process;  
sending a request for a lock object for the distributed resource to a first local lock manager process of the distributed lock manager process, the request including the home location;  
receiving the lock object for the distributed resource from a second local lock manager process executing on the unique host, if a number of outstanding locks granted by the second local lock manager process is less than a value of a local resource maximum defined for the second local lock manager process;  
and  
providing access to the resource to the first client only in response to receiving the lock object.

44. An apparatus controlling concurrent users of a distributed resource on a network, wherein the resource is limited to a maximum number of concurrent users, comprising:  
a host that is communicatively coupled to a network that contains the distributed resource and that comprises one or more processors and a computer-readable storage medium;  
one or more sequences of instructions stored in the computer-readable storage medium which, when executed by the one or more processors, cause the one or more processors to carry out the steps of:  
generating a value for a local resource maximum number of users stored on each host of a plurality of hosts such that a summation over the plurality of hosts of the value for the local resource maximum yields an aggregate value that does not exceed the maximum number of concurrent users;  
determining whether to increase a first value in a first resource maximum stored on a first host of the plurality of hosts; and

15 if it is determined to increase the first resource maximum, then  
16 decreasing by a particular amount a second value in a second resource  
17 maximum stored on a second host of the plurality of hosts, and  
18 increasing by the particular amount the first value in the first resource  
19 maximum stored on the first host,  
20 granting a lock for the resource if the number of outstanding locks granted by a lock  
21 manager process is less than a value of the local resource maximum stored on  
22 the first host.

THESE